



THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of

NEVILL et al.

Atty. Ref.: 550-192

Serial No. 09/731,060

Group: 2194

Filed: December 7, 2000

Examiner: Zhen, Li B.

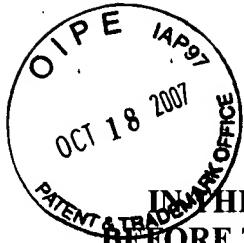
For: SCHEDULING CONTROL WITHIN A SYSTEM HAVING MIXED
HARDWARE AND SOFTWARE BASED INSTRUCTION
EXECUTION

Before the Board of Patent Appeals and Interferences

REPLY BRIEF FOR APPELLANT

**On Appeal From Final Rejection
From Group Art Unit 2194**

John R. Lastova
NIXON & VANDERHYE P.C.
11th Floor, 901 North Glebe Road
Arlington, Virginia 22203-1808
(703) 816-4025
Attorney for Appellants
Nevill et al and
ARM Limited



THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Patent Application of

NEVILL et al.

Atty. Ref.: 550-192

Serial No. 09/731,060

TC/A.U.: 2194

Filed: December 7, 2000

Examiner: Zhen, Li B.

For: SCHEDULING CONTROL WITHIN A SYSTEM HAVING MIXED
HARDWARE AND SOFTWARE BASED INSTRUCTION EXECUTION

October 18, 2007

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

REPLY BRIEF

Sir:

This is a reply to selected points made in the Examiner's Answer dated August 20, 2007.

Appellants' Brief demonstrated that Envoy does not return control to the hardware based execution unit for a next instruction to be executed. By arguing on page 12 of the Answer that the claims do not require switching from one execution mode to another execution mode, the Examiner misses the point. The switching between execution modes is relevant because that is what Envoy discloses: switching between native and platform independent modes (see, e.g., 4:53-64). The teachings of Envoy must be read in context and as a whole.

The Examiner relies on column 7, lines 15-28 of Envoy as teaching "program instructions...forwarded to said software based execution unit for execution with control being returned to said hardware based execution unit for a next program instruction to be executed." This reliance is misplaced because the Examiner is ignoring the context in which these lines must be understood. Envoy's multiplexer 56 only selects the next byte code when the Envoy

system operates in a platform-independent mode. But the Evoy system is in a native mode, i.e., not in the platform-independent mode, when an instruction is executed using software-based execution.

Starting at the bottom of column 6, line 61, Evoy explains that a switch has occurred from native mode to platform independent mode. But when the processor encounters a non-native bytecode instruction it cannot handle while in the platform-independent mode, software interpretation of the bytecode is performed in the native mode. So in that native mode, the processor accesses native instructions to implement the complex non-native bytecode instruction. See 5:64-67 and 7:27-29. When the processor is in its normal native mode, the statements regarding “selecting the next bytecode” (7:20) do not apply. In native mode, the next native instruction is selected and not the next non-native bytecode. Thus, contrary to the Examiner’s assertion, 7:8-16 of Evoy does not disclose the claimed feature where control is returned to the hardware based execution unit for a next program instruction to be executed. This fundamental misunderstanding of Evoy undermines several of the Examiner’s arguments presented in the Answer.

On page 13 of the Examiner’s Answer, the Examiner contends that the claims do not recite preventing scheduling of operations from being inappropriately triggered part-way through the software interpretation and that, as a consequence, the claims fail to distinguish from the timer-based scheduling of the prior art. Appellants disagree. Claim 1 recites “program instructions to be executed are sent to said hardware based execution unit for execution” and “control being returned to said hardware based execution unit for a next program instruction to be executed” in the event that a previous instruction has been forwarded to the software based execution unit for execution. Together, these features prevent scheduling operations from being inappropriately triggered. By routing all of the program instructions through the hardware-based

execution unit, it can keep track of the execution of instructions and accordingly generate a scheduling signal for triggering a scheduling operation irrespective of whether the preceding instructions have been executed by hardware or software.

At the bottom of page 13 of the Answer, the Examiner contends that Gee's scheduling of threads so that the highest priority thread is dispatched teaches "triggering a scheduling operation to be performed between program instructions for managing scheduling between threads or tasks irrespective of whether a preceding program instruction was executed by said hardware based execution unit or said software based execution unit." From the context provided by the other features of claim 1, it is clear that the scheduling logic feature (v) of claim 1 "irrespective of whether..." is linked to keeping track of execution of all instructions by always routing program instructions to be executed to the hardware based execution unit and also returning control to the hardware based execution unit regardless of where the program instructions were executed.

It is significant that neither Gee nor Evoy recognizes the problem solved by the claimed technology which avoids not only inappropriate triggering of scheduling operations but also the disadvantageous overhead of counter exchange in processing systems having mixed hardware-based execution units and software-based execution units.

For the reasons explained above and in the Brief, the Board should reverse the final rejection.

NEVILL et al.
Serial No. 09/731,060
October 18, 2007

Respectfully submitted,

NIXON & VANDERHYE P.C.

By:



John R. Lastova
Reg. No. 33,149

JRL:maa
901 North Glebe Road, 11th Floor
Arlington, VA 22203
Telephone: (703) 816-4000
Facsimile: (703) 816-4100